

**PATENT APPLICATION**

**AUTOMATIC TESTING FOR PROGRAMMABLE NETWORKS OF  
CONTROL SIGNALS**

Inventor: Jayabrata Ghosh Dastidar, a citizen of India, residing at  
984 Kiely Blvd. #K  
Santa Clara, CA 95051

Assignee: Altera Corporation  
101 Innovation Drive  
San Jose, CA, 95134

Entity: Large

## AUTOMATIC TESTING FOR PROGRAMMABLE NETWORKS OF CONTROL SIGNALS

### BACKGROUND OF THE INVENTION

5 [0001] This application relates to techniques for automatically testing circuits, and more particularly, to techniques for testing programmable networks of control signals.

[0002] Field programmable gate arrays (FPGAs) are programmable integrated circuits that include logic elements, memory, and programmable interconnect conductors. The logic elements, memory, and the interconnect conductors can be programmed to implement a  
10 number of different user designs. The logic elements typically include combinatorial and sequential logic circuits.

[0003] After an FPGA is manufactured, the logic elements, memory, and programmable connections in the interconnect structure are tested to ensure that they are operating properly. Tests are performed to detect the presence of any manufacturing defects in the FPGA.

15 [0004] Programmable interconnects that route control signals are tested functionally, by observing their effect on functional registers. For example, all programmable interconnections that can connect a clock signal to a clock input of a functional register are tested one at a time. Functional registers are registers that can be programmed and operated during user mode. The tests are performed to determine if the clock signal clocked the  
20 function register as expected. If the functional register was not clocked correctly, the programmable interconnections or the clock path may contain defects that prevent proper operation of the functional register.

[0005] It is expensive to test all of the programmable interconnections that constitute the programmable network for the control signals in this way, because a large number of  
25 configurations is needed to test all of the interconnections. Because the tests are functional and the same registers are used for observation, a clock connection cannot be tested at the same time as a clear connection on the same register.

[0006] Therefore, it would be desirable to provide more efficient and cost effective techniques for testing networks of programmable resources that route control signals through  
30 a programmable integrated circuit.

## BRIEF SUMMARY OF THE INVENTION

[0007] The present invention provides techniques for automatically testing resources that route control signals on programmable integrated circuits (ICs). Programmable ICs have a programmable interconnect structure that connects programmable logic elements, memory, and other on-chip circuit elements.

[0008] The techniques of the present invention can be used to test networks of programmable interconnections that route control signals to registers on a programmable IC. According to the present invention, control signals (such as clock signals) are routed through a logic gate to a dedicated test register. Values of the control signals are stored in the test register, transmitted outside the IC, and then compared to expected values to identify defects in the programmable interconnections.

[0009] An enable circuit couples the control signals to functional registers on the programmable IC during user mode. The enable circuit decouples the control signals from the functional registers during test mode, so that the control signals do not interfere with tests of the functional registers. During the test procedures of the present invention, the control signals are treated as data signals and are not used to control other registers on the IC. The control signals are only routed to the dedicated test register during test mode.

[0010] Other objects, features, and advantages of the present invention will become apparent upon consideration of the following detailed description and the accompanying drawings, in which like reference designations represent like features throughout the figures.

## BRIEF DESCRIPTION OF THE DRAWINGS

[0011] Figure 1 is a block diagram of a system for testing programmable interconnections that route control signals on a programmable IC according to an embodiment of the present invention;

[0012] Figure 2 is a simplified block diagram of a programmable logic device that can implement embodiments of the present invention; and

[0013] Figure 3 is a block diagram of an electronic system that can implement embodiments of the present invention.

## DETAILED DESCRIPTION OF THE INVENTION

[0014] Figure 1 is a block diagram that illustrates a test system for a programmable IC according to an embodiment of the present invention. Programmable ICs include FPGAs, programmable logic devices (PLDs), programmable logic arrays (PLAs), configurable logic  
5 arrays, etc.

[0015] Programmable integrated circuits (ICs) contain a programmable routing structure that includes rows and columns of interconnect conductors. The interconnect conductors can be programmably coupled together (or decoupled) using programmable interconnection  
10 circuits. The interconnect conductors and programmable interconnection circuits are referred to as routing resources.

[0016] Most of the programmable routing resources are used to route data signals within a programmable IC. However, a dedicated subset of the programmable routing resources are used to route control signals within programmable ICs. For example, a clock tree is  
15 a dedicated set of routing resources that is used to route clock signals applied to input pins on an IC. In a programmable IC, the clock tree routing resources are programmable.

[0017] Programmable ICs also include routing resources that route control signals between programmable logic elements. In Altera's Stratix family of FPGAs, sets of logic elements and other circuit elements are grouped into logic array blocks (LABs). The routing resources that route signals to LABs are referred to as LAB lines in Figure 1.

[0018] According to the present invention, an automatic test generator creates test paths to  
20 test programmable routing resources that route control signals. The test paths start from clock input pins, input/output (I/O) pins, or source registers. The test paths are routed through the programmable routing resources, through a logic gate, and end at dedicated test registers.

[0019] The test generator also generates test vectors that can test the test paths for structural  
25 defects such as stuck-at defects, continuity defects, and bridging faults. The present invention can test routing resources that route control signals such as clock signals, clear signals, clock enable signals, and other types of control signals.

[0020] Figure 1 illustrates an example of a test path of the present invention. A test  
30 generator of the present invention can generate a test path originating from a clock input pin or from a source register in a LAB. Test paths originating from a clock pin are connected to

routing multiplexers 111 through the clock tree inputs as shown in Figure 1. Test paths originating from LABs are connected to routing multiplexers 111 through the LAB lines.

[0021] Routing multiplexers 111 are routing resources that route control signals through the test paths to XOR gate 112 as shown in Figure 1. XOR gate 112 applies a very well-known exclusive OR Boolean function to its input signals. Test register 114 is the destination register of a test path. Register 114 is a dedicated test register that is only used for testing the routing resources that route the control signals. Test register 114 is not programmed or operated during user mode. Test register 114 stores the output signal of XOR gate 112 on each rising edge of the Test Clock signal applied to its clock input.

[0022] In the example of Figure 1, the test generator routes four control signal through test paths from a clock pin or source register through XOR gate 112 to test register 114. Clock/clear signals CLK0, CLK1, CLR0, and CLR1 are examples of four control signals that can be tested by the present invention. All four clock/clear signals CLK0, CLK1, CLR0, and CLR1 are routed to inputs of XOR gate 112.

[0023] The test generator of the present invention creates test vectors to test the programmable routing resources. The test vectors are applied sequentially to the clock pins and the source registers. The values of signals CLK0, CLK1, CLR0, and CLR1 vary with each new test vector applied to a source register or clock pin. For each binary value of the four input control signals, XOR gate 112 generates a unique output signal value. Each output signal value is applied to the data input D of test register 114 and stored briefly in test register 114.

[0024] Test register 114 sequentially stores a series of output values generated by XOR gate 112 in response to the different test vectors. Test register 114 then transmits each value generated by XOR 112 outside the IC. An external test system compares the XOR gate generated values to expected values to identify any defects in routing multiplexers 111 and any other control signal routing resources. If some of the signals from XOR gate 112 do not match expected values, a defect in routing multiplexers 111 or any other control signal routing resources is a possible cause.

[0025] Routing multiplexers 111 can be reprogrammed to test another set of programmable interconnections. A new set of four control signals are applied to inputs of XOR gate 112, stored in register 114, and tested externally.

[0026] Every control signal on a programmable IC can be tested using the circuitry of Figure 1. After a sufficient amount of test data has been accumulated, routing resources containing defects can be isolated by a process of elimination using well-known techniques. By determining which sets of test vectors lead to erroneous output values from XOR gate 112, a test system can isolate the location of defects in the routing resources.

[0027] The control signals tested by XOR gate 112 are used to control numerous registers on the programmable IC during normal operation of the circuit. These registers are represented by functional registers 115 in Figure 1. The functional registers are typically registers inside logic elements on the programmable IC that are programmed and operated in the user mode. During the user mode, the programmable IC is configured according to a user made design.

[0028] The functional registers 115 are also tested for defects after production of the programmable IC. Functional registers 115 are coupled in test paths. The tests can be controlled by the test clock signal.

[0029] When the functional registers are tested during test mode, enable circuit 113 decouples the control signals (CLK0, CLK1, CLR0, CLR1) from functional registers 115 so that registers 115 and other routing resources can be tested independently from the control signals. Enable circuit 113 couples the control signals to functional registers 115 during the user mode of the programmable IC. Enable circuit 113 has an output signal line for each control signal input. A handful of functional patterns can test the paths from the control signals through enable circuit 113 to functional registers 115.

[0030] The present invention provides techniques for testing networks of routing resources that route control signals on a programmable IC. The control signals are treated as data signals during the test procedure. The control signals are decoupled from the functional registers 115 that they control during user mode. Values of the control signals are sampled by a dedicated test register 114 after passing through a logic gate as discussed above to detect defects in the routing resources.

[0031] The present invention substantially reduces the costs associated with testing interconnections (e.g., by 50%). The present invention also simplifies test generation. The present invention has a negligible effect on die size of the integrated circuit, because the only additional circuitry that needs to be added to an FPGA is XOR gate 112 and test register 114.

The techniques of the present invention are also less susceptible to test mode glitches. The present invention also allows for additional structural testing like testing for bridging faults.

[0032] An XOR gate is used in Figure 1 as an example. Other logic gates that implement other Boolean functions (e.g., AND or OR Boolean functions) can also be used in alternative embodiments of the present invention. Also, XOR gate 112 can be substituted with a multiplexer. In other embodiments, XOR gate 112 can receive more or less than 4 control signals.

[0033] Figure 2 is a simplified partial block diagram of an exemplary high-density PLD 300 wherein techniques of the present invention can be utilized. PLD 300 includes a two-dimensional array of programmable logic array blocks (or LABs) 302 that are interconnected by a network of column and row interconnects of varying length and speed. LABs 302 include multiple (e.g., 10) logic elements (or LEs), an LE being a small unit of logic that provides for efficient implementation of user defined logic functions.

[0034] PLD 300 also includes a distributed memory structure including RAM blocks of varying sizes provided throughout the array. The RAM blocks include, for example, 512 bit blocks 304, 4K blocks 306 and a MegaBlock 308 providing 512K bits of RAM. These memory blocks can also include shift registers and FIFO buffers. PLD 300 further includes digital signal processing (DSP) blocks 310 that can implement, for example, multipliers with add or subtract features. I/O elements (IOEs) 312 located, in this example, around the periphery of the device support numerous single-ended and differential I/O standards. It is to be understood that PLD 300 is described herein for illustrative purposes only and that the present invention can be implemented in many different types of PLDs, FPGAs, and the like.

[0035] While PLDs of the type shown in Figure 2 provide many of the resources required to implement system level solutions, the present invention can also benefit systems wherein a PLD is one of several components. Figure 3 shows a block diagram of an exemplary digital system 400, within which the present invention can be embodied. System 400 can be a programmed digital computer system, digital signal processing system, specialized digital switching network, or other processing system. Moreover, such systems can be designed for a wide variety of applications such as telecommunications systems, automotive systems, control systems, consumer electronics, personal computers, Internet communications and networking, and others. Further, system 400 can be provided on a single board, on multiple boards, or within multiple enclosures.

[0036] System 400 includes a processing unit 402, a memory unit 404 and an input/output (I/O) unit 406 interconnected together by one or more buses. According to this exemplary embodiment, a programmable logic device (PLD) 408 is embedded in processing unit 402. PLD 408 can serve many different purposes within the system in Figure 3. PLD 408 can, for example, be a logical building block of processing unit 402, supporting its internal and external operations. PLD 408 is programmed to implement the logical functions necessary to carry on its particular role in system operation. PLD 408 can be specially coupled to memory 404 through connection 410 and to I/O unit 406 through connection 412.

[0037] Processing unit 402 can direct data to an appropriate system component for processing or storage, execute a program stored in memory 404 or receive and transmit data via I/O unit 406, or other similar function. Processing unit 402 can be a central processing unit (CPU), microprocessor, floating point coprocessor, graphics coprocessor, hardware controller, microcontroller, programmable logic device programmed for use as a controller, network controller, and the like. Furthermore, in many embodiments, there is often no need for a CPU.

[0038] For example, instead of a CPU, one or more PLDs 408 can control the logical operations of the system. In an embodiment, PLD 408 acts as a reconfigurable processor, which can be reprogrammed as needed to handle a particular computing task. Alternately, programmable logic device 408 can itself include an embedded microprocessor. Memory unit 404 can be a random access memory (RAM), read only memory (ROM), fixed or flexible disk media, PC Card flash disk memory, tape, or any other storage means, or any combination of these storage means.

[0039] While the present invention has been described herein with reference to particular embodiments thereof, a latitude of modification, various changes, and substitutions are intended in the present invention. In some instances, features of the invention can be employed without a corresponding use of other features, without departing from the scope of the invention as set forth. Therefore, many modifications can be made to adapt a particular configuration or method disclosed, without departing from the essential scope and spirit of the present invention. It is intended that the invention not be limited to the particular embodiments disclosed, but that the invention will include all embodiments and equivalents falling within the scope of the claims.